



Universidade do Minho
Licenciatura em Engenharia Informática

Unidade Curricular de Bases de Dados

Ano Lectivo de 2023/2024

Secret Story – Agência de Detetives

**Edgar(A99890), Miguel(A104451),
Joshua(A105684), Silvério(A101536),
Oleksii(A102131) – Grupo 02**

Abril, 2024

BD

Data de Recepção	
Responsável	
Avaliação	
Observações	

Secret Story – Agência de Detetives

**Edgar(A99890), Miguel(A104451),
Joshua(A105684), Silvério(A101536),
Oleksii(A102131) – Grupo 02**
Abril, 2024

Resumo

Este relatório foi realizado no âmbito da unidade curricular de Bases de Dados, que tinha como objetivo principal desenvolver um sistema de base de dados, de forma avaliar os métodos, organização, trabalho em equipa e os nossos conhecimentos sobre a unidade curricular.

De um modo geral, o nosso projeto consiste na estruturação e construção de uma base de dados de uma agência de detetives imaginária, neste caso, a *Secret Story*, que seja capaz de suportar a informação necessária para uma melhor organização, mais facilidade e simplicidade nas tarefas da agência, seguindo determinados padrões de modelação e implementação de acordo com Connolly e Begg (2015) – **Bibliografia**.

Antes de entrar na implementação física da base de dados, foi registada a informação necessária da agência, apresentado o seu contexto, o porquê da necessidade de uma base de dados e planeamento geral do projeto. De seguida, passamos para o levantamento dos requisitos, de forma a ser possível a construção do modelo conceptual na fase seguinte, que nos dará um rascunho daquilo que iremos construir. Utilizando numa outra fase “MySQL Workbench”, teremos a criação de um modelo lógico, a partir do qual geraremos um modelo físico, que convergirá juntamente com a implementação de queries e do modelo físico, para a base de dados final, dando-se por terminado o projeto.

As fases descritas anteriormente passarão por revisões e validações antes de serem implementadas.

Área de Aplicação: Sistemas de Base de Dados

Palavras-Chave: Base de dados, Dados, Agência, Detetive, SQL, Modelo Conceptual, Modelo Lógico, Entidades, Relacionamentos, Atributos

Índice

1. Definição de Sistema	1
1.1. Contexto de Aplicação	1
1.2. Motivação e Objetivos do Trabalho.....	2
1.3. Análise da Viabilidade do Processo	3
1.4. Recursos e Equipa de Trabalho.....	3
1.5. Plano de Execução.....	5
2. Levantamento e Análise dos Requisitos	6
2.1. Método de Levantamento e de Análise de Requisitos Adotado	6
2.2. Organização dos Requisitos Levantados.....	7
2.2.1. Requisitos de Descrição	7
2.2.2. Requisitos de Manipulação	8
2.2.3. Requisitos de Controlo	9
2.3. Análise e Validação Geral dos Requisitos	9
3. Modelação Conceptual	10
3.1. Apresentação da Abordagem de Modelação Realizada.....	10
3.2. Identificação e Caracterização das Entidades	10
3.3. Identificação e Caracterização dos Relacionamentos	11
3.4. Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos.	12
3.5. Apresentação e Explicação do Diagrama ER Produzido.....	13
4. Modelação Lógica	13
4.1. Construção e Validação do Modelo de Dados Lógico	13
4.2. Apresentação e Explicação do Modelo Lógico Produzido	14
4.3. Normalização de Dados	17
4.4. Validação do Modelo com Interrogações do Utilizador	18
5. Implementação Física	22
5.1. Apresentação e explicação da base de dados implementada.....	22
5.2. Criação de utilizadores da base de dados	23
5.3 Povoamento da base de dados.....	26
5.4 Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)	28
5.5. Definição e caracterização de vistos de utilização em SQL	31
5.6. Tradução das interrogações do utilizador para SQL.....	32
5.7. Indexação do Sistema de Dados	35
5.8. Implementação de procedimentos, funções e gatilhos	36
6. Conclusões e Trabalho Futuro	40
Siglas	41
Bibliografia	41
Anexos	41

Índice de Figuras

Figura 1 - Diagrama de Gantt inicial	5
Figura 2 - Modelo Conceptual	13
Figura 3 - Modelo Lógico	16
Figura 4 - Árvore da 1 ^a interrogação	18
Figura 5 - Árvore da 2 ^a interrogação	19
Figura 6 - Árvore da 3 ^a interrogação	20
Figura 7 - Árvore da 4 ^a interrogação	21
Figura 8 - Criação da Base de Dados e Criação da tabela Centro	22
Figura 9 - Exemplo de criação de um Utilizador e respetivo papel.....	23
Figura 10 - Código de criação da "ROLE" Diretores e suas permissões	23
Figura 11 - Código da criação da "ROLE" dos clientes e suas permissões	24
Figura 12 - Código de Criação de "ROLE" dos agentes e suas permissões	25
Figura 13 - Código de inserção de Centros	26
Figura 14 - Código de inserção de Clientes	26
Figura 15 - Código de associação de um Tipo com um Caso	27
Figura 16 - Excerto de código do programa de inserção.....	27
Figura 17 - Código da Vista "CasosEmProgressoPorAgente".....	31
Figura 18 - Código SQL da 1 ^a Interrogação	32
Figura 19 - Código SQL da 2 ^a Interrogação	33
Figura 20 - Código SQL da 3 ^a Interrogação	34
Figura 21 - Código SQL da 4 ^a Interrogação	35
Figura 22 - Criação do índice idxEMail associado ao "email" de "Cliente"	35
Figura 23 - Criação do índice idxDataDenuncia associado à "data_denuncia" de "Denuncia"	36
Figura 24 - Criação e Input do procedimento "addNovoCaso"	36
Figura 25 - Código de execução do procedure "addNovoCaso"	37
Figura 26 - Código da função "contarCasosResolvidosAgente" e exemplos de uso	38
Figura 27 - Código do gatilho "atualizarCargoAgente"	39

Índice de Tabelas

Tabela 1 - Recursos	3
Tabela 2 – Equipa de Trabalho.....	4
Tabela 3 – Requisitos de Descrição	7
Tabela 4 – Requisitos de Manipulação	8
Tabela 5 – Requisitos de Controlo	9
Tabela 6 – Caraterização das Entidades.....	10
Tabela 7 – Caraterização dos Relacionamentos	11
Tabela 8 – Caraterização dos Atributos	12
Tabela 9 - Cálculo de espaço da tabela Centro.....	28
Tabela 10 - Cálculo de espaço da tabela Agente	28
Tabela 11 - Cálculo de espaço da tabela Cliente	28
Tabela 12 - Cálculo de espaço da tabela Denúncia	29
Tabela 13 - Cálculo de espaço da tabela Caso.....	29
Tabela 14 - Cálculo de espaço da tabela Tipo de caso	29
Tabela 15 - Cálculo de espaço da tabela Evidencia.....	29
Tabela 16 - Cálculo de espaço da tabela Suspeito	30
Tabela 17 - Cálculo de espaço da tabela Telemóvel do cliente.....	30
Tabela 18 - Cálculo de espaço da tabela Agente responsável por caso	30
Tabela 19 - Cálculo de espaço da tabela Caso apresenta suspeito.....	30
Tabela 20 - Cálculo de espaço da tabela Caso pertence a tipo	30

1. Definição do Sistema

1.1. Contexto de aplicação

A agência de detetives “*Secret Story*” (**ST**) localiza-se no distrito de Braga, contendo o centro de operações no interior da Universidade do Minho e espalhando-se por Portugal continental através de uma extensa rede de túneis, que ligam os diversos centros. Com a queda do Estado Novo, o sistema viu-se obrigado a criar uma agência privada diretamente conectada à organização intergovernamental “*INTERPOOL*” no dia 25 de abril de 1977, após a fundação da universidade.

Além do centro principal na universidade bracarense, a agência privada expandiu o seu trabalho do norte ao sul do país, com mais locais de trabalho e recrutamento nas diversas zonas. Com o avanço da tecnologia e modernização, estes centros são frequentemente remodelados e atualizados com novos equipamentos e mais documentação, de forma a melhorar o trabalho dos agentes e consequentemente ajudar na proteção da sociedade portuguesa a desvendar os maus que circulam o país.

Os detetives têm como “cabecilha” o diretor Orlando “*Holmes*”, antigo agente da “*INTERPOOL*”, que geriu a agência desde a sua fundação até ao tempo atual e conta com a sua atual equipa de subdiretores a quem deposita toda a confiança, formada por Vasco, André, Diana, Marisa e Regina (responsáveis pelos restantes 5 centros secundários de operações em Lisboa, Faro, Aveiro, Bragança e Castelo Branco respetivamente), mais os restantes agentes, que necessitam de autorização e recrutamento especial para pertencerem à atividade, de forma a serem os mais indicados e capazes de tratar de casos de todos os tipos.

Até há 6 meses, a **ST** trabalhava apenas com casos especiais e altamente confidenciais entregues pelos altos escalões do governo, mas, visto que, o tipo de casos desses baixou, o diretor Orlando decidiu abrir portas para um novo projeto, uma agência pública que tratasse dos casos pedidos pelo povo e dos seus segredos. Desde então, a agência foi expandindo aos poucos, aumentando o número de agentes e o número de casos dos diversos centros espalhados pelo país, o que originou um caos na papelada nestes últimos meses. Tanto o diretor como os subdiretores estavam com dificuldades em gerir os diversos centros, não conseguindo, por exemplo, controlar os agentes e os seus casos de forma organizada e rápida, nem distribuir os mesmos, tendo ainda outros problemas como controlar o registo de novos casos e a sua clientela.

Para colocar um ponto final nesse caos, o diretor Orlando decidiu parar o funcionamento da mesma por uns tempos e reunir os seus subdiretores, que sugeriram implementar uma base de dados para a sua agência, o que não agradou ao professor Orlando que tanto gostava dos seus documentos em papel, mas que se viu obrigado a concordar, de forma a gerir a agência da melhor forma. Sendo assim, as denúncias por telefonemas ou entregues nos centros de operações serão agora passadas para um site e respetiva base de dados onde cada diretor/subdiretor do seu centro de operações registará as denúncias para casos e distribuirá pelos seus detetives.

1.2. Motivação e Objetivos do Trabalho

Baseado na sua experiência como diretor e pelas deficiências do seu sistema atual, o Diretor Orlando decidiu definir uma série de metas que precisam ser alcançados com essa implementação para que se torne possível a substituição do sistema atual para esse sistema de bases de dados, nomeadamente:

- Organizar e separar os seus casos, bem como otimizar o processo de entrega e fechamento dos mesmos, de forma a aumentar a organização.
- Registar informação dos clientes e agentes, para facilitar o contacto com os mesmos e resolução dos casos.
- Monitorar os agentes, de forma a saber quais casos tais agentes estão a trabalhar e o seu progresso respetivo.
- Fácil acesso. Por ser da velha guarda, o Diretor Orlando “*Holmes*” ainda possui muitas dificuldades quando se trata de acesso a tecnologia, por esse facto necessita de um sistema intuitivo para todos os detetives, clientes e diretores.
- Gerir de forma efetiva as denúncias feitas, em particular todas aquelas que se relacionam com o público geral.

1.3. Análise da Viabilidade do processo

Com base no estudo realizado na fase preliminar deste projeto conseguiu-se levantar objetivos a atingir com esta base dados e os seus possíveis impactos:

- A implementação da base de dados requer um investimento inicial de €50.000 em desenvolvimento e infraestrutura. Espera-se que, após a implementação, haja um retorno anual de €20.000 em custos operacionais, devido à redução de tempo e recursos necessários para gerir os casos manualmente.
- Está quase a chegar a época balnear o que leva a um aumento de turismo e consequentemente de denúncias. Esperamos implementar a base de dados antes disso para suavizar o impacto da época alta na eficiência da agência.
- Atualmente leva cerca de 4 horas para arquivar um caso depois de resolvido, esta base de dados poderá reduzir esse tempo para menos de 1 hora (pois permite a atualização dos dados mais fácil e rápida durante o caso, diminuindo a quantidade de dados a registar).
- Reduzir os erros cometidos semanalmente no registo de casos em 80% passando de 5 para apenas um.
- Os casos são distribuídos manualmente, levando em média 2 horas por caso, enquanto, com a base de dados, se espera reduzir esse tempo para meia hora. Isso representa uma redução de 75% no tempo de distribuição de casos.
- Aumentar a taxa de satisfação do cliente devido a uma resposta mais rápida e eficiente dos casos.

1.4. Recursos e Equipa de Trabalho

Materiais	Humanos
<ul style="list-style-type: none">• Os Documentos em papel da agência<ul style="list-style-type: none">• Computadores e SharePoint• Software (<i>BrModelo</i> e <i>MySQLWorkbench</i>)	<ul style="list-style-type: none">• Cabecilha Orlando• A equipa dos subdiretores de todos os distritos constituída por Vasco, André, Diana, Marisa e Regina<ul style="list-style-type: none">• Agentes• Equipa de 5 programadores<ul style="list-style-type: none">• Clientes da agência

Tabela 1 - Recursos Necessários

Expansão na utilidade de cada recurso

- Os documentos em papel da agência são cruciais para a coleta de dados. Desde a obtenção de horários de todos os membros da agência e a sua identidade, até os tipos de casos e os resultados dos casos em que trabalharam individualmente ou em equipa no passado, para mais facilmente distribuir os casos futuros de forma mais organizada e rápida, e importantes também para acessar melhor as necessidades de todos os centros secundários de operações e produzir resultados que resolvem as necessidades de toda a agência.
- Os computadores e SharePoint são cruciais para os informáticos coletarem dados e partilharem entre todos os analistas a informação necessária para a estruturação da base de dados.
- O *BrModelo* será necessário para fazer um esquema conceptual de “esboço” relativa à estrutura da base de dados que nos ajudará na construção do esquema lógico construído através do *MySQLWorkbench*.
- O Cabecilha Orlando é necessário como diretor para expor as necessidades gerais que a agência enfrenta, traçar os objetivos que a agência pretende atingir com a base de dados em funcionamento, e transmitir para nós como a agência é gerida e como funciona, para assim criarmos uma base de dados que se encaixe melhor nas engrenagens da agência.
- A equipe de subdiretores dos diversos concelhos, constituída por Vasco, André, Diana, Marisa e Regina, é crucial para nos transmitir as necessidades da agência em nível local e mais preciso, além de nos fornecer um contato mais direto com os agentes e o seu dia a dia.
- Os agentes são parte dos nossos recursos, pois são eles que serão capazes de informar sobre suas necessidades únicas e as da agência no seu dia a dia, além de melhor explicar como uma gestão mais eficiente dos horários e uma melhor distribuição de cada caso para cada tipo de agente pode ser alcançada, para que cada agente trabalhe nos casos em que está melhor qualificado.
- A equipe de 5 programadores será responsável por trabalhar em colaboração com a agência para coletar os dados disponíveis, dar sentido a essas informações e criar a base de dados necessária.

Pessoal Interno	Pessoal Externo
<ul style="list-style-type: none"> • Cabecilha Orlando • A equipa dos subdiretores de todos os distritos constituída por Vasco, André, Diana, Marisa e Regina <ul style="list-style-type: none"> • Agentes • Clientes da agência 	<ul style="list-style-type: none"> • 5 analistas/programadores
<ul style="list-style-type: none"> • Fornecer dados • Expor preocupações para se resolver na agência • Ajudar na composição dos horários • Tratar do funcionamento da agência <ul style="list-style-type: none"> • Resolver casos 	<ul style="list-style-type: none"> • Coleção de dados • Aprender como a agência funciona <ul style="list-style-type: none"> • Levantamento de requisitos • Criar soluções para a resolução dos problemas <ul style="list-style-type: none"> • Modelação do sistema • Explorar os recursos disponíveis para criação da base de dados <ul style="list-style-type: none"> • Construção da base de dados

Tabela 2 - Equipa de trabalho e suas tarefas

1.5. Plano de Execução do Projeto

De forma a termos uma ideia das datas/durações possíveis para cada etapa e, consequentemente facilitar na organização e divisões da equipa de trabalho, decidimos criar um plano/cronograma ilustrativo, mais especificamente um Diagrama de “Gantt”, através do programa *Excel*, que nos ajuda a visualizar e programar as tarefas necessárias para a realização do projeto relativo à base de dados da agência.

Juntamente com os Diretores da Agência de Detetives “*Secret Story*”, a equipa de analistas criou o seguinte plano:

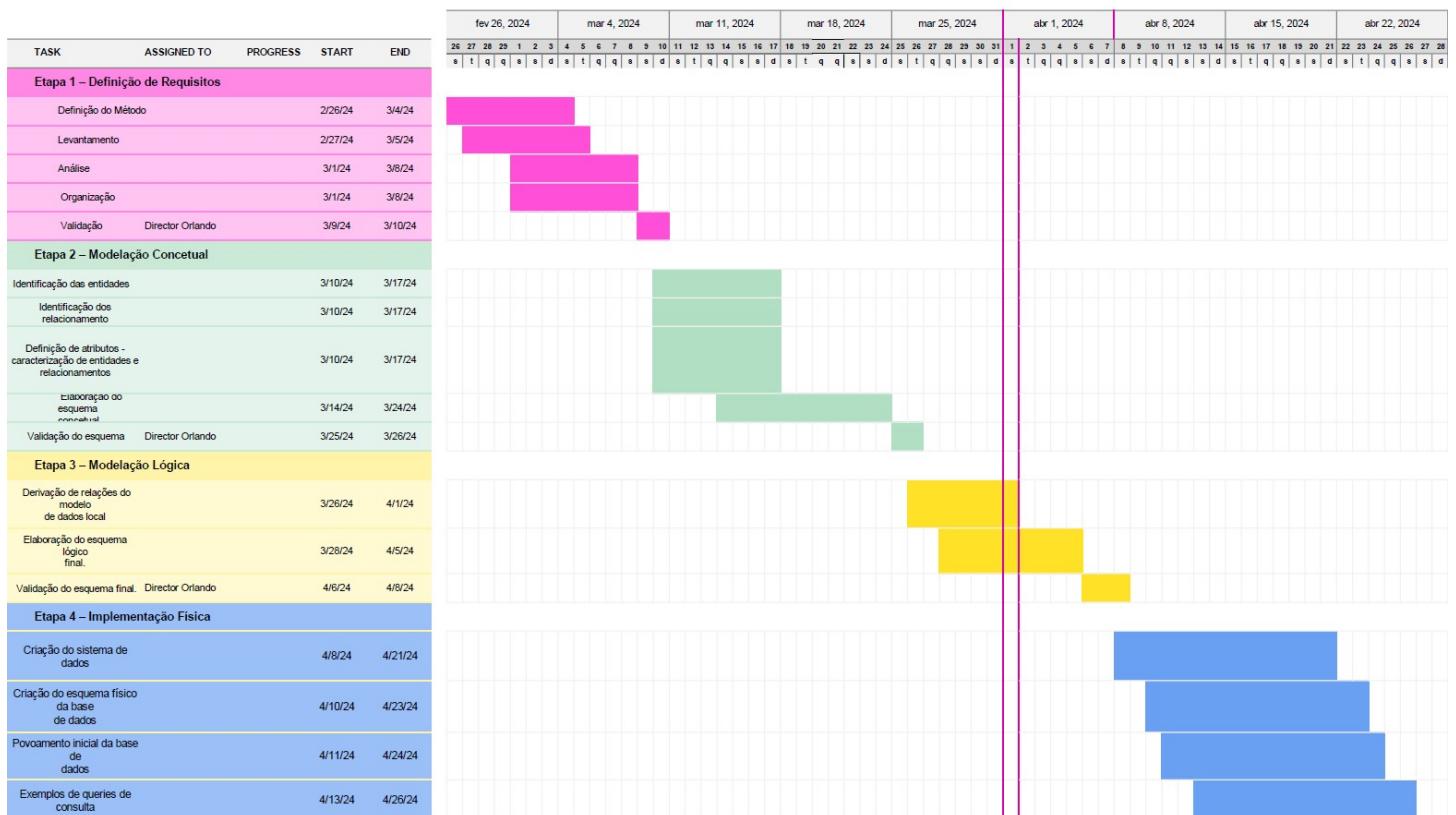


Figura 1 - Diagrama de Gantt inicial

2. Levantamento e Análise de Requisitos

2.1. Método de Levantamento e de Análise de Requisitos Adotado

Para um melhor entendimento dos dados, funcionalidades e outros requisitos pretendidos na base de dados da agência, procedeu-se ao levantamento dos mesmos da seguinte forma:

- Analisando a documentação referente, maioritariamente, à informação dos agentes, clientes e casos dos diferentes centros - disponibilizada pelo Diretor Orlando e Subdiretores, para separar a informação útil da restante.
- Reunindo, na primeira semana de março, com o diretor e certos subdiretores, de forma a obter a opinião de cada um sobre o sistema anterior e melhorias pretendidas;
- Acompanhando missões de certos agentes recomendados pelo diretor Orlando, de modo a conhecer o seu método de trabalho e respetivas necessidades.
- Inquerindo sobre a opinião de alguns agentes e clientes, relativamente aos seus respetivos casos e experiências na agência, com o fim de melhorar a experiência futura.

Parte da entrevista ao agente Joaquim na tarde de 9 de março:

“(...)

Pergunta do analista Edgar: - E já agora, você tem tido algumas dificuldades nas suas missões, ou tem algo que queira mudar relativamente ao sistema?

Resposta do agente Joaquim: Não são muitas as queixas, mas que me facilitava ter acesso direto a dados do cliente e aos seus casos anteriores, que podem até ter alguma relação com os próximos, claro que me facilitava. Gostava que houvesse um método mais eficaz para o fazer, porque até agora, tenho ido chatear sempre o meu chefe Orlando e ele ainda tem a trabalheira de procurar pelos arquivos, que demora sempre um bocadinho. (...)"

Parte da entrevista a um cliente que pediu anonimato, na manhã do dia 10 de março:

“(...)

Pergunta do analista Edgar: - Outra questão, como classifica a rapidez e qualidade do serviço até agora? O que mudava?

Resposta do cliente: Não posso dizer que foi rápida nem o melhor serviço, mas o caso foi resolvido, por isso saio satisfeita, apesar de me darem a notícia horrível do meu marido andar com outra. Ouvi dizer que agora vai ser tudo feito num site ou base não sei quê, por isso sugiro que haja uma forma de os clientes verificarem o progresso feito no caso, para sabermos novidades sem estarmos constantemente a contactar o detetive ou a central. E além disso, também podiam deixar os clientes alterar a sua informação do site, que eu estou sempre a mudar o número de telemóvel.

(...)"

No **Anexo 1** é disponibilizada uma parte de uma Ata de uma reunião com os diretores.

2.2. Organização dos Requisitos Levantados

A partir das entrevistas, reuniões e outros métodos de levantamento de informação referidos e demonstrados anteriormente, recolheu-se alguns requisitos da agência, sendo os essenciais para o melhoramento da agência posteriormente distribuídos e organizados em três tipos de requisitos, com o respetivo número de identificação, data e hora da recolha, a sua descrição, a área referente, a fonte da informação (os diretores, clientes, agentes) e o nome do Analista que o registou.

2.2.1. Requisitos de Descrição

Requisitos de Descrição				
Nº	Data/Hora	Descrição	Área	Fonte
RD01	08/03/2024 10:15H	A um caso tem de estar associado a um agente ou mais agentes, a uma denúncia, apresentando o id do caso, um ou mais tipos de casos, a data de início e de fim, o seu estado atual (Em progresso, Encerrado) e um relatório.	Casos	Diretor Orlando Miguel
RD02	08/03/2024 10:22H	A uma denúncia tem de estar associada a um identificador, uma localização (concelho, código postal, rua), uma data, um cliente, uma descrição e um estado (Não registada ou Registada).	Denúncias	Subdiretora Diana Miguel
RD03	08/03/2024 10:31H	Cada cliente deve ser identificado por um número único e nome, a sua morada (rua, concelho, código postal), email e palavra-passe da plataforma, tendo pelo menos um número telefónico associado.	Cliente	Subdiretor Vasco Silvério S.
RD04	08/03/2024 14:40H	Cada agente é identificado por um número único, nome e cargo(Novato, Perito, Veterano, Ex-Detetive), a sua morada (rua,concelho, código postal), a data de nascimento,email e palavra-passe da plataforma, tendo um número telefónico associado.	Agente	Diretor Orlando Miguel
RD05	08/03/2024 15:00H	Uma evidência depende de um único caso e é caracterizada por um identificador, uma data de recolha e descrição relativa ao que aconteceu no caso.	Casos	Subdiretora Marisa Joshua D.
RD06	09/03/2024 16:30H	Cada tipo de caso tem associado um número e um nome e engloba vários casos.	Casos	Subdiretora Marisa Silvério S.
RD07	09/03/2024 16:34H	Um suspeito pode estar relacionado a um ou mais casos e está associado a um número, a um nome e descrição do sujeito.	Casos	Agente José Edgar
RD08	09/03/2024 18:39H	Um centro de operações deve estar associado a um número de identificação, a uma localização (Rua, concelho, código postal) e ao subdiretor/diretor responsável, juntamente com o número telefónico do centro.	Centro	Diretor Orlando Oleksii
RD09	09/03/2024 18:50H	Um caso pode ou não ter diversas evidências e suspeitos associados.	Casos	Subdiretor André Edgar
RD10	10/03/2024 09:50H	O agente pertence a um único centro de operações, enquanto este último tem vários agentes associados.	Agente	Subdiretora Regina Edgar
RD11	11/03/2024 09:30H	Uma denúncia pode ser registada como um caso, posteriormente.	Casos	Subdiretora Diana Oleksii
RD12	11/03/2024 10:00H	Um agente pode estar encarregue de um, muitos ou nenhum caso.	Agente	Subdiretora Diana Oleksii
RD13	11/03/2024 11:00H	Um cliente tanto pode fazer nenhuma como uma ou mais denúncias, mas esta pertence a apenas um único cliente.	Cliente	Subdiretora Regina Edgar
RD14	11/03/2024 11:30H	Um suspeito pode participar de um ou diversos casos.	Casos	Subdiretora Regina Edgar

Tabela 3 - Requisitos de Descrição

2.2.2. Requisitos de Manipulação

Requisitos de Manipulação					
Nº	Data/Hora	Descrição	Área	Fonte	Analista
RM01	08/03/2024 10:05H	Criar, gerir e visualizar clientes e os seus dados.	Cliente	Subdiretor André	Joshua D.
RM02	08/03/2024 11:15H	Criar, gerir e visualizar agentes e os seus dados.	Agente	Subdiretora Regina	Joshua D.
RM03	08/03/2024 11:21H	Visualizar casos associados a um determinado agente ou cliente.	Casos	Subdiretora Diana	Silverio S.
RM04	09/03/2024 11:00H	Saber nº de denúncias feitas por certos clientes, entre datas ou numa determinada localização.	Denúncias	Subdiretora Diana	Edgar
RM05	09/03/2024 11:05H	Filtrar caso pelos tipos (homicídio, rapto, etc.).	Casos	Subdiretor André	Silverio S.
RM06	09/03/2024 11:10H	Num determinado momento, saber o nº de casos resolvidos para um dado agente.	Agente	Subdiretor André	Silverio S.
RM07	09/03/2024 11:11H	Promover ou despromover agentes.	Agente	Subdiretor Vasco	Joshua D.
RM08	09/03/2024 11:30H	Registrar, gerir e visualizar casos.	Casos	Diretor Orlando	Silverio S.
RM09	10/03/2024 08:58H	Registrar, gerir e visualizar denúncias.	Denúncias	Subdiretora Marisa	Joshua D.
RM10	10/03/2024 09:02H	Filtrar agentes de um certo cargo.	Agente	Diretor Orlando	Joshua D.
RM11	10/03/2024 09:05H	Adicionar, gerir e visualizar evidências.	Casos	Subdiretora Marisa	Joshua D.
RM12	10/03/2024 09:06H	Enumerar informação dos agentes responsáveis por um caso.	Casos	Subdiretora Regina	Edgar
RM13	10/03/2024 10:01H	Visualizar casos que começaram/terminaram numa/entre uma determinada data.	Casos	Subdiretora Regina	Oleksii
RM14	10/03/2024 10:15H	Registrar, gerir e visualizar novos suspeitos.	Casos	Subdiretora Regina	Silverio S.
RM15	10/03/2024 10:18H	Enumerar casos de um certo tipo.	Casos	Subdiretor André	Edgar
RM16	11/03/2024 10:21H	Listar casos por ordem de data de entrada ou por estado atual.	Casos	Subdiretor Vasco	Silverio S.
RM17	11/03/2024 12:20H	Inserir, gerir e visualizar tipos de caso.	Casos	Subdiretor Diana	Edgar
RM18	11/03/2024 14:15H	Visualizar casos com um certo estado (Encerrado, Em progresso).	Casos	Subdiretor André	Joshua D.
RM19	11/03/2024 14:16H	Inserir, gerir e visualizar centros de operações.	Centro	Diretor Orlando	Edgar
RM20	11/03/2024 14:20H	Enumerar/Contar agentes em cada centro de operações da agência.	Agente	Subdiretor Vasco	Edgar
RM21	11/03/2024 14:30H	Pesquisar suspeitos, clientes e agentes pelo nome.	Geral	Subdiretor Vasco	Edgar
RM22	11/03/2024 15:30H	Enumerar suspeitos/evidências relacionados com um caso.	Casos	Subdiretor Marisa	Edgar

Tabela 4 - Requisitos de Manipulação

2.2.3. Requisitos de Controlo

Requisitos de Controlo					
Nº	Data/Hora	Descrição	Área	Fonte	Analista
RC01	08/03/2024 10:20H	O cliente pode visualizar os dados dos casos somente relacionados às suas denúncias.	Cliente	Subdiretor Vasco	Miguel
RC02	08/03/2024 10:20H	O agente pode alterar a informação dos casos.	Agente	Subdiretor Vasco	Miguel
RC03	08/03/2024 11:25H	Os diretores têm permissão de acesso, de inserção e de modificação nos dados de todas as entidades.	Geral	Subdiretor Andre	Joshua D.
RC04	08/03/2024 11:30H	O agente pode aceder a todos os casos guardados.	Casos	Subdiretor Vasco	Joshua D.
RC05	09/03/2024 11:30H	O agente pode visualizar os dados dos clientes.	Cliente	Agente Joaquim	Edgar
RC06	10/03/2024 11:37H	Somente os diretores podem atribuir casos.	Casos	Subdiretora Diana	Silverio S.
RC07	11/03/2024 10:15H	Somente os diretores podem adicionar e/ou eliminar o acesso a um caso.	Casos	Subdiretor Vasco	Silverio S.
RC08	11/03/2024 10:16H	Somente os diretores podem adicionar agentes.	Agente	Subdiretor André	Joshua D.
RC09	11/03/2024 10:18H	O agente pode aceder e inserir dados das evidências e suspeitos.	Casos	Subdiretor Vasco	Joshua D.
RC10	11/03/2024 10:32H	Somente os diretores podem promover os agentes.	Agente	Subdiretora Diana	Silverio S.
RC11	11/03/2024 11:34H	O cliente pode modificar, visualizar e inserir a informação do seu perfil.	Cliente	Cliente da Agência	Edgar
RC12	11/03/2024 12:15H	Os agentes podem modificar e atualizar a informação dos suspeitos.	Casos	Agente Emanuel	Edgar
RC13	12/03/2024 12:30H	Todos têm acesso de visualização à informação relativa aos centros de operações.	Centro	Cliente Ermegildo	Edgar
RC14	12/03/2024 12:40H	Somente os diretores podem inserir novos tipos de casos.	Casos	Diretor Orlando	Oleksii
RC15	12/03/2024 12:45H	Somente os diretores podem registrar novos centros e modificar a sua informação.	Centros	Diretor Orlando	Edgar

Tabela 5 - Requisitos de Controle

2.3. Análise e Validação Geral dos Requisitos

Após o levantamento e análise dos requisitos foi necessário realizar uma análise e validação geral dos mesmos. Para isso, primeiro a nossa equipa reuniu-se com o intuito de analisar e verificar a viabilidade, redundância e objetividade dos mesmos. Verificou-se que estava tudo em concordância.

De seguida, apresentamos os requisitos aos Diretores/Subdiretores, de forma a validarmos os mesmos, que foram examinados e verificados com sucesso, satisfazendo os diretores.

Por fim, ajustamos os requisitos, de forma a passar para a próxima fase do projeto.

3. Modelação Conceptual

3.1. Apresentação da Abordagem de Modelação Realizada

Tendo os requisitos validados e bem definidos, passamos para a fase de modelação conceptual, onde planearemos um “esboço” da nossa base de dados, de forma a nos oferecer uma percepção da sua estrutura, que nos ajudará na continuação do projeto.

Como processo de modelação conceptual, optamos por um Diagrama **ER** (Entidade-Relacionamento), que demonstra, através de um desenho conceptual, as entidades, os seus relacionamentos e respetivos atributos, sendo de fácil compreensão. Antes de iniciar a construção do esquema conceptual, é importante identificar e caracterizar as entidades, relacionamentos e atributos obtidos através dos requisitos de descrição anteriormente definidos, de forma a facilitar as posteriores ilustrações.

Relativamente à Agência de Detetives do Diretor Orlando, iniciamos de seguida o processo referido com a construção de tabelas em *Excel* com os dados necessários de acordo com *Connolly e Begg*, seguidas de um esquema conceptual desenhado no *BrModelo*, ferramenta adotada para a construção do Diagrama ER.

3.2 Identificação e Caracterização das Entidades

Relativamente às entidades, primeiramente, identificamos as mesmas, descrevendo-as e demonstrando alguns sinónimos. Além disso, referimos quando/como as mesmas serão inseridas na base de dados, identificando, por fim o requisito da **Tabela 3** que deu origem a esta caraterização e identificação:

Caraterização das entidades				
Nome da entidade	Descrição	Sinónimo	Ocorrência	Origem
Cliente	Uma pessoa ou entidade que faz uma denúncia ou solicita assistência à agência de detetives. Armazena informações de contacto do cliente tais como email, morada, nome e números telefónicos e também a sua palavra passe.	Público, Utente	Iniciado quando um cidadão se cadastrá, sendo inserido na base de dados.	RD03
Agente	Uma pessoa que é funcionária do Centro de Detetives e que desempenha cargo de agente. Armazena as informações de Contato, cargo e informações de funcionário.	Detetive	Iniciado quando o diretor adiciona o agente à base de dados.	RD04
Denúncia	Uma queixa feita por um cliente relativa à um acontecimento que possui potencial de infringir a lei. Armazena os dados de data, descrição e local.	Queixa, Alerta, Comunicação, Notificação	É criada quando um cliente regista um acontecimento na plataforma.	RD02
Caso	Um infringimento da lei ou algum acontecimento que necessita de investigação para ser concluída. Armazena o estado, descrição e datas.	Incidente, Crime, Evento	É criado quando um diretor valida uma denúncia.	RD01
Centro	Local físico onde se encontram agentes em expediente e seus respectivos diretores/subdiretores. Armazena nome da direção do local, localização, e telefone de contato.	Delegacia, Sede	É registado à base de dados quando se é fundada uma nova delegacia.	RD08
Evidência	Unidade de informação referente ao caso à qual pode auxiliar na resolução de um caso. Armazena apenas uma breve descrição e data de recolha.	Prova, Vestigio, Testemunho	É iniciada quando um agente registra uma evidência.	RD05
Suspeito	Uma pessoa acusada de realizar um crime. Armazena um nome e uma descrição.	Suspeitos	É registrado na base de dados quando um agente recebe a informação do suspeito.	RD07
Tipo de caso	Categorização dos casos por características chaves referentes a um crime. Armazena um nome.	Categoria do Caso	É registrado quando um sub/diretor insere um novo tipo de caso, advindo das denúncias.	RD06

Tabela 6 - Caraterização das Entidades

3.3. Identificação e Caracterização dos Relacionamentos

Relativamente aos relacionamentos entre as diversas entidades, caraterizamos os mesmos tendo em conta as respetivas entidades, a multiplicidade de cada entidade relativamente à outra, o nome do relacionamento, terminando com uma descrição/significado do relacionamento e o requisito originário da **Tabela 3**.

Optamos por utilizar apenas relacionamentos binários, sem atributos nos mesmos.

Caraterização dos relacionamentos						
Entidade	Multiplicidade	Relacionamento	Multiplicidade	Entidade	Descrição	Origem
Cliente	(1,1)	regista	(0,n)	Denúncia	Um único cliente regista nenhuma, uma ou mais denúncias.	RD13
Caso	(1,n)	apresenta	(0,n)	Suspeito	Um ou mais casos apresentam nenhum, um ou mais suspeitos.	RD14;RD09
Agente	(1,n)	pertence a	(1,1)	Centro	Um ou mais agentes pertencem obrigatoriamente a um único centro de operações.	RD10
Agente	(1,n)	responsável por	(0,n)	Caso	Um ou mais agentes são responsáveis por nenhum, um ou mais casos.	RD12;RD01
Caso	(1,n)	pertence a	(1,n)	Tipo de Caso	Um ou mais casos pertencem a um ou mais tipos/categorias de caso.	RD01;RD06
Caso	(1,1)	tem	(0,n)	Evidência	Um único caso tem nenhuma, uma ou mais evidências associadas.	RD05;RD09
Denúncia	(1,1)	é transformada em	(0,1)	Caso	Uma única denúncia é transformada/registada em nenhum ou um único caso.	RD11; RD1

Tabela 7 - Caraterização dos Relacionamentos

3.4. Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos

Relativamente aos atributos das entidades (sem atributos nos relacionamentos), identificamos e descrevemos brevemente os mesmos, referindo o tipo de dado, se é multivalorado, se pode ser nulo, e o respetivo requisito da **Tabela 3**:

Caraterização dos atributos						
Entidade	Atributos	Descrição	Tipos de dados	Multivalorado	Null	Origem
Cliente	Nome de Usuário	Nome documentado no cartão de cidadão do cliente	VARCHAR	Não	Não	RD03
	Id	Identificador Único do Cliente	INT	Não	Não	RD03
	Número Telefónico	Número associado ao telemóvel fornecido pelo cliente	CHAR	Sim	Não	RD03
	Email	Endereço de email fornecido pelo cliente	VARCHAR	Não	Não	RD03
	Palavra Passe	Senha de acesso à conta do cliente	VARCHAR	Não	Não	RD03
	Morada	Localização da residência do Cliente				RD03
	Concelho (Mor.)	Concelho onde habita o Cliente	VARCHAR	Não	Não	RD03
	Rua (Mor.)	Rua onde habita o Cliente	VARCHAR	Não	Não	RD03
	Código Postal (Mor.)	Código Postal do Cliente (do tipo 0000-000)	CHAR	Não	Não	RD03
Agente	Nome de Usuário	Nome documentado no cartão de cidadão do agente	VARCHAR	Não	Não	RD04
	Id	Identificador Único do Agente	INT	Não	Não	RD04
	Número Telefónico	Número associado ao telemóvel fornecido pelo centro	CHAR	Não	Não	RD04
	Email	Endereço de email fornecido pelo centro	VARCHAR	Não	Não	RD04
	Cargo	Cargo exercido pelo agente (Novato, Perito, Veterano, Ex-Detetive)	VARCHAR	Não	Não	RD04
	Data de Nascimento	Data de Nascimento do agente	DATE	Não	Não	RD04
	Palavra Passe	Senha de acesso à conta do Agente	VARCHAR	Não	Não	RD04
	Morada	Localização da residência do Agente				RD04
	Concelho (Mor.)	Concelho onde habita o Agente	VARCHAR	Não	Não	RD04
Denúncia	Rua (Mor.)	Rua onde habita o Agente	VARCHAR	Não	Não	RD04
	Código postal (Mor.)	Código Postal do Agente (do tipo 0000-000)	CHAR	Não	Não	RD04
	Id	Identificador único da denúncia	INT	Não	Não	RD02
	Data	Data do registo do crime	DATE	Não	Não	RD02
	Descrição	Breve resumo do acontecimento que levou à denúncia	VARCHAR	Não	Não	RD02
	Estado	Estado onde se encontra a Denúncia (Registada ou Não registada)	VARCHAR	Não	Não	RD02
	Localização	Local onde o suposto crime aconteceu				RD02
	Concelho (Loc.)	Concelho onde ocorreu o potencial caso	VARCHAR	Não	Não	RD02
	Rua (Loc.)	Rua onde ocorreu o potencial caso	VARCHAR	Não	Não	RD02
Caso	Código Postal (Loc.)	Código Postal do potencial caso (do tipo 0000-000)	CHAR	Não	Não	RD02
	Id	Identificador Unico do Caso	INT	Não	Não	RD01
	Data de Inicio	Data de quando o caso é registado	DATE	Não	Não	RD01
	Data de Fim	Data na qual o caso passa de "Em Progresso" para "Encerrado"	DATE	Não	Sim	RD01
	Estado Atual	Estado do caso (Em progresso, Encerrado)	VARCHAR	Não	Não	RD01
Evidência	Relatório	Relatório de caso, com o progresso feito até o momento ("Sem informação" se não houver progresso ainda registado)	VARCHAR	Não	Não	RD01
	Id	Identificador único da evidência	INT	Não	Não	RD05
	Data de recolha	Data de recolha da amostra/evidência	VARCHAR	Não	Não	RD05
	Descrição	Breve descrição da evidência sobre o uso dela dentro do caso	VARCHAR	Não	Não	RD05
Suspeito	Id	Identificador único do suspeito	INT	Não	Não	RD07
	Nome	Nome documentado no cartão de cidadão do Suspeito	VARCHAR	Não	Não	RD07
	Descrição	Descrição com atributos do suspeito	VARCHAR	Não	Não	RD07
Tipo de Caso	Nome	Nome associado a um tipo de crime (Homicídio, Roubo, Violência Doméstica, etc.)	VARCHAR	Não	Não	RD06
	Número	Número de identificação do tipo de caso	INT	Não	Não	RD06
Centro	Id	Identificador único do centro de operações	INT	Não	Não	RD08
	Direção	Nome do Diretor/Sub diretor	VARCHAR	Não	Não	RD08
	Localização	Local onde se encontra o centro de operações				RD08
	Concelho (Loc.)	Concelho onde se encontra o centro	VARCHAR	Não	Não	RD08
	Rua (Loc.)	Rua onde se encontra o Centro	VARCHAR	Não	Não	RD08
	Código Postal (Loc.)	Código Postal do Centro (do tipo 0000-000)	CHAR	Não	Não	RD08
	Telefone	Número do telefone do centro de operações	CHAR	Não	Não	RD08

Tabela 8 - Caraterização dos Atributos

3.5. Apresentação e Explicação do Diagrama ER Produzido

Tendo em conta os requisitos obtidos e uma análise dos mesmos, juntamente com a identificação das entidades, relacionamentos e atributos assinalados anteriormente, foi possível construir um esquema conceptual ou Diagrama ER (Entidade-Relacionamento) da base de dados (usando a ferramenta BrModelo), com o qual podemos visualizar uma estrutura inicial que nos ajudará nas próximas fases do projeto:

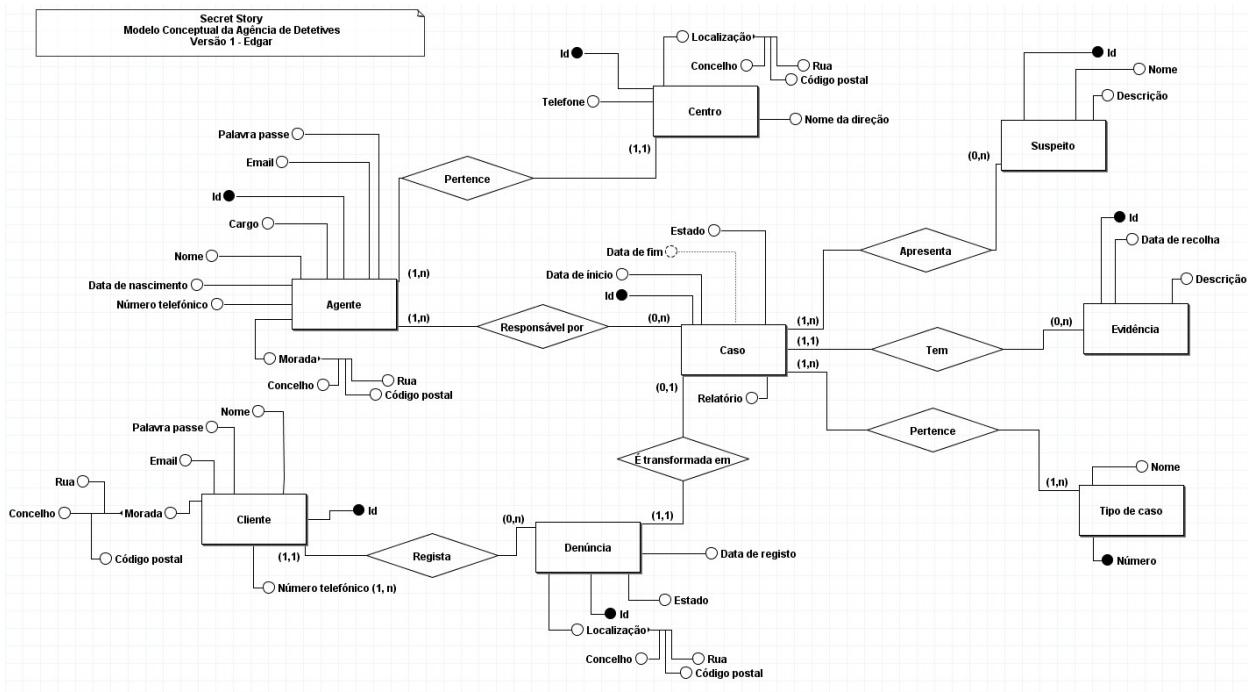


Figura 2 - Modelo Conceptual

Examinando as caracterizações nos capítulos 3.2., 3.3. e 3.4. e este esquema do Modelo Conceptual, concluímos que tanto as entidades, como os seus respetivos atributos e relacionamentos estão corretos e em concordância.

4. Modelação Lógica

4.1. Construção e Validação do Modelo de Dados Lógico

Partindo do nosso modelo conceptual anteriormente gerado e validado, com as corretas identificações e caracterizações de entidades, relacionamentos e atributos, passamos para uma modelação lógica, onde essas entidades, relacionamentos e atributos farão parte de um conjunto de tabelas.

Utilizando as regras de conversão associadas, o modelo lógico produzido terá doze tabelas, das quais oito serão relativas a cada entidade, três serão resultado dos relacionamentos N para N e a última será resultado de um atributo mencionado anteriormente como multivalorado.

4.2. Apresentação e Explicação do Modelo Lógico Produzido

Assim, começando pelas entidades, as tabelas serão constituídas pelos seguintes dados:

➤ **Agente:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** centro_id (INT);
- **Restantes Atributos:** nome (VARCHAR (30)), cargo (VARCHAR (15)), data_nasc (DATE), email (VARCHAR (50)), pass (VARCHAR (15)), concelho (VARCHAR (20)), rua (VARCHAR (50)), cod_postal (CHAR (8)), tele (CHAR(13));

➤ **Cliente:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** inexistente;
- **Restantes Atributos:** nome (VARCHAR (30)), email (VARCHAR (50)), pass (VARCHAR (15)), concelho (VARCHAR (20)), rua (VARCHAR (50)), cod_postal (CHAR (8));

➤ **Denúncia:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** cliente_id (INT);
- **Restantes Atributos:** data (DATE), estado_atual (VARCHAR (15)), concelho (VARCHAR (20)), rua (VARCHAR (50)), cod_postal (CHAR (8)), descricao (VARCHAR (150));

➤ **Caso:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** denuncia_id (INT);
- **Restantes Atributos:** data_registro (DATE), data_final (DATE, podendo ser nula), estado_atual (VARCHAR (15)), relatorio (VARCHAR (300));

➤ **Centro:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** inexistente;
- **Restantes Atributos:** nome_direcao (VARCHAR (10)), concelho (VARCHAR (20)), rua (VARCHAR (50)), cod_postal (CHAR (8)), tele (CHAR(13));

➤ **Suspeito:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** inexistente;
- **Restantes Atributos:** nome (VARCHAR (30)), descricao (VARCHAR (150));

➤ **Tipo de caso:**

- **Chave primária:** numero (INT);
- **Chave estrangeira:** inexistente;
- **Restantes Atributos:** nome (VARCHAR (30));

➤ **Evidência:**

- **Chave primária:** id (INT);
- **Chave estrangeira:** caso_id (INT);
- **Restantes Atributos:** data_recolha (DATE), descricao (VARCHAR (150));

Agora, relativamente aos relacionamentos:

➤ **Agente-Caso:**

- **Chave primária:** composta pelas duas chaves agente_id e caso_id (INT);
- **Chave estrangeira:** caso_id (INT), agente_id (INT);
- **Restantes Atributos:** inexistentes;

➤ **Caso-Suspeito:**

- **Chave primária:** composta pelas duas chaves suspeito_id e caso_id (INT);
- **Chave estrangeira:** caso_id (INT), suspeito_id (INT);
- **Restantes Atributos:** inexistentes;

➤ **Caso-Tipo de caso:**

- **Chave primária:** composta pelas duas chaves caso_id e tipo_numero (INT);
- **Chave estrangeira:** caso_id (INT), tipo_numero (INT);
- **Restantes Atributos:** inexistentes;

Por fim, a tabela relativa aos números de telemóveis dos clientes:

➤ **Telemóveis dos clientes:**

- **Chave primária:** tele_cliente (CHAR(13));
- **Chave estrangeira:** cliente_id (INT);
- **Restantes Atributos:** inexistentes;

Tendo em conta as diferentes vistas das diversas tabelas descritas mais a junção dos respetivos relacionamentos e respeitando as regras de criação do modelo lógico através do modelo conceptual, obtemos o seguinte esquema lógico (realizado no programa MySQL Workbench):

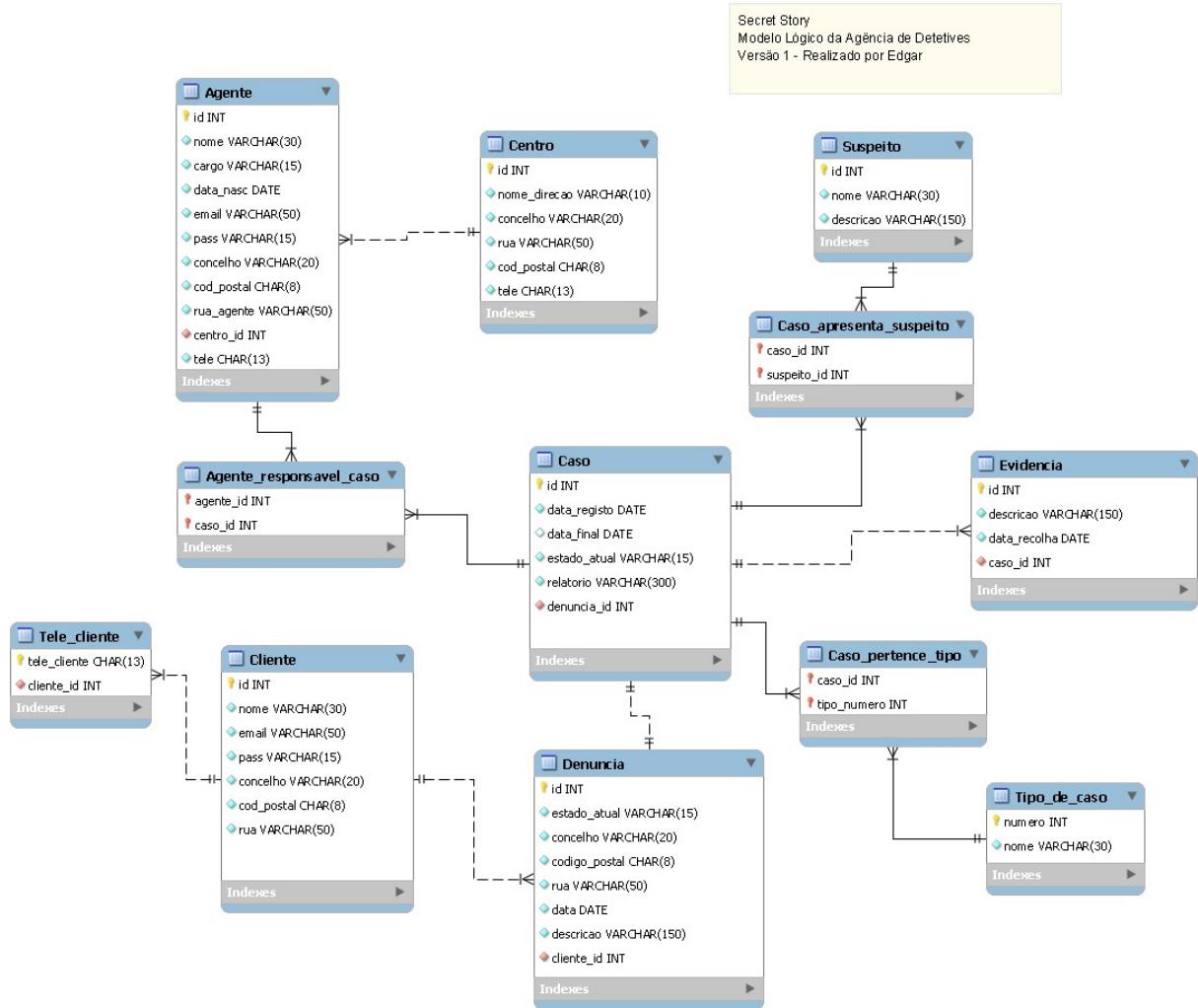


Figura 3 - Modelo Lógico

4.3. Normalização de Dados

Contribuindo para a validação do modelo lógico realizado, é importante ver se o mesmo está normalizado. Tendo em conta as três regras de normalização descritas por *Connolly e Begg*, verificamos o seguinte:

- Através da **Primeira Fórmula Normal** (1FN), sabemos que os atributos não podem se repetir, ou apresentarem mais que um valor nas respetivas tabelas, o que não acontece no nosso modelo lógico.
- Já na **Segunda Fórmula Normal** (2FN), atributos normais dependem apenas da chave primária da respetiva tabela, sendo válida para todas as tabelas do nosso modelo.
- Por último, na **Terceira Fórmula Normal** (3FN), nenhum atributo deve depender de outro atributo, o que é verdadeiro para todos os atributos do nosso modelo.

Tendo em conta esta análise, concluímos que o nosso modelo está normalizado.

4.4. Validação do Modelo com Interrogações do Utilizador

De forma a concluirmos a validação do modelo lógico, criamos algumas interrogações de utilizador com uso da álgebra relacional e alguns dos requisitos de manipulação anteriormente estabelecidos.

A. Contar quantas denúncias os clientes submeteram – RM04

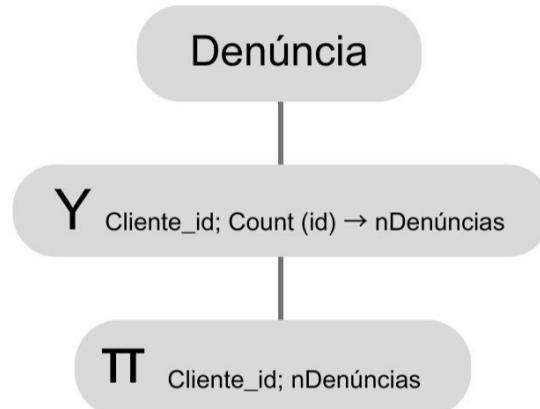


Figura 4 - Árvore da 1^a interrogação

Primeiramente começamos por selecionar a entidade das Denúncias, após isso contamos todos os ids distintos e agregamos esse número em uma variável chamada “nDenúncias”, por fim projetamos tal valor.

B. Listar quais agentes estão associados a um determinado caso – RM12

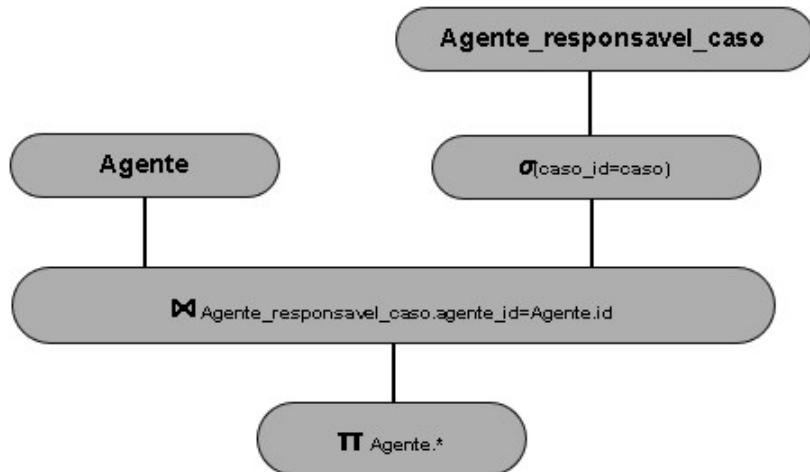


Figura 5 - Árvore da 2^a interrogação

Primeiro selecionamos a relação entre o agente e o caso, conhecida como “Agente_responsavel_caso” e faz-se um filtro para encontrar os casos pretendido. De seguida, será feita um junção da seleção feita e da tabela agente para recolhermos a informação dos agentes associados ao caso respetivo, cuja informação será projetada.

C. Contar número de agentes dos diversos centros – RM20

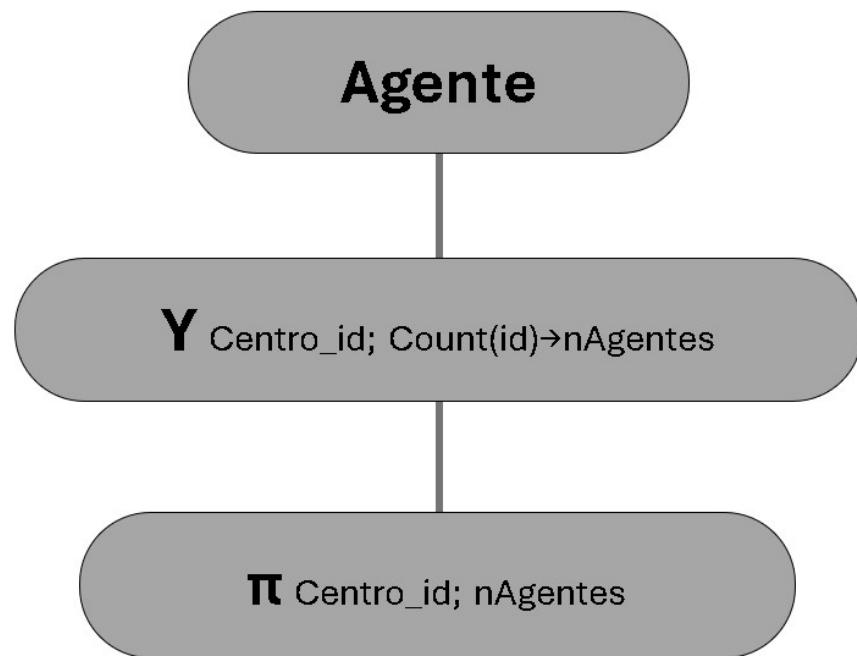


Figura 6 - Árvore da 3^a interrogação

Primeiro selecionamos a tabela relativa à entidade “Agente”, após isso contamos o número de agentes em cada centro (ou número de id's de agentes) que apresentam o mesmo “centro_id” (chave estrangeira). Esse número é atribuído à variável “nAgentes”, que será projetada por fim junto com o respetivo centro_id.

D. Listar todos os casos resultantes de denúncias de um cliente – RM03

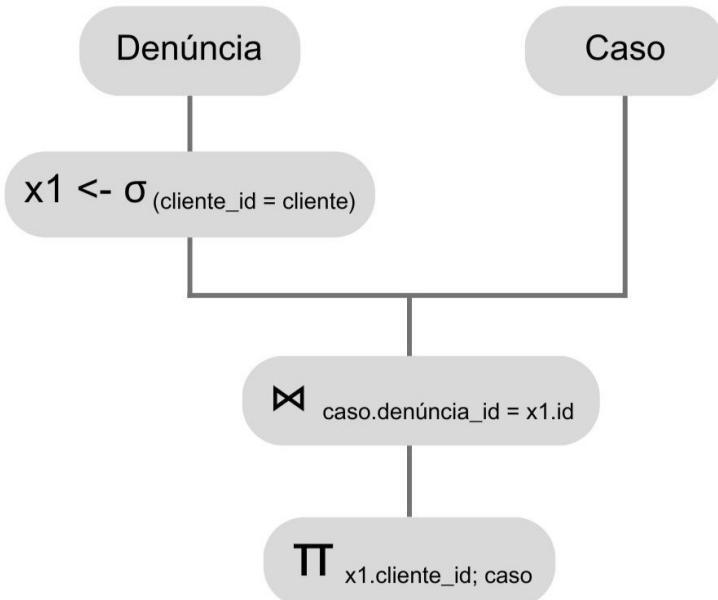


Figura 7 - Árvore da 4^a interrogação

Ao selecionar a entidade Denúncia temos acesso tanto aos ids dos clientes como que dos ids das denúncias, então utilizamos a função de seleção para filtrar todos as denúncias feitas por um cliente e então utilizamos o id das denúncias para comparar as denúncias com os casos. Ao final projetamos todas as informações do caso e o id do Cliente.

5. Implementação Física

5.1 Apresentação e explicação da base de dados implementada

Passando para a parte final do projeto, a implementação física da Base de Dados da Agência, foi necessário o uso de um DBMS, neste caso, o **MySQL**, que juntamente com o **MySQLWorkbench**, nos permitiu construir fisicamente a base de dados e gerir a mesma.

Com o modelo lógico feito, prosseguimos para a criação da Base de Dados (com o nome ‘Agencia’) e criação das respetivas tabelas de acordo com as tabelas da modelação lógica, transferidas agora para código **SQL**, ou seja, físicas, com os “*CREATE SCHEMA*” e “*CREATE TABLE*”. A primeira tabela a ser criada foi a dos registos dos centros de operações ‘Centro’, visto que não tinha nenhuma chave estrangeira. De seguida criamos a dos agentes Agente, com a chave estrangeira a fazer referência ao ‘id’ da tabela ‘Centro’. E fomos seguindo este mesmo método com as restantes tabelas (tendo especial atenção à necessidade das tabelas e suas chaves estrangeiras), ‘Cliente’, ‘Denuncia’, ‘Tipo_de_caso’, ‘Caso’, ‘Suspeito’, ‘Evidencia’, terminando com as tabelas geradas dos relacionamentos, com as devidas características, já referidas no **Capítulo 4**. O código correspondente à criação das mesmas estará presente no *script* enviado no **Anexo 2**.

```
12      -- Criação da base de dados
13 •  CREATE SCHEMA IF NOT EXISTS Agencia DEFAULT CHARACTER SET utf8 ;
14 •  USE Agencia;
15      -- DROP DATABASE Agencia;
16
17      -- Centro
18 •  CREATE TABLE IF NOT EXISTS Centro (
19          id INT AUTO_INCREMENT UNIQUE NOT NULL PRIMARY KEY,
20          nome_direcao VARCHAR(40) NOT NULL,
21          concelho VARCHAR(20) NOT NULL,
22          rua VARCHAR(50) NOT NULL,
23          cod_postal CHAR(8) NOT NULL,
24          tele CHAR(13) NOT NULL
25      );
26      -- DROP TABLE Centro;
```

Figura 8 - Criação da Base de Dados e Criação da tabela Centro

Na criação das tabelas, foi utilizado o “*AUTO_INCREMENT*” em todos os ids, de modo a automaticamente, incrementar o atributo indicado como chave primária na criação de um registo, facilitando a posterior inserção dos registo. Também foi aproveitada a funcionalidade do “*DEFAULT*” como por exemplo para datas automáticas com “*CURRENT_DATE*” na criação de registo com datas. Com isto, temos o primeiro passo da implementação física.

5.2 Criação de utilizadores da base de dados

Para a criação de Utilizadores e o seu papel na base de dados, na recém-criada Base de Dados, foi utilizado o “CREATE USER” e “CREATE ROLE”. Depois serão garantidas as diversas permissões (estabelecidas na **Secção 2**, nos Requisitos de Controlo) com base na sua “ROLE”, usando o “GRANT”. Todos códigos correspondentes estarão presentes no *script* enviado no **Anexo 3**.

```
-- Criação de user
CREATE USER 'user' IDENTIFIED BY 'password';
GRANT 'diretores' TO 'diretor_user';
GRANT 'agentes' TO 'agente_user';
GRANT 'clientes' TO 'cliente_user';
```

Figura 9 – Exemplo de criação de um Utilizador e respetivo papel

Na **Figura 9** podemos observar que para o ‘user’ foi atribuído um identificador chamado de ‘password’ que representa a palavra-passe do utilizador. Após termos definido o ‘user’ utilizamos o “CREATE ROLE” para gerir permissões baseado no tipo do utilizador (Diretor, agente, cliente).

- Diretores

```
-- Criação de uma função para a direção
CREATE ROLE 'diretores';
-- Garantir todas as permissões para estes
GRANT ALL ON Agencia TO 'diretores';
```

Figura 10 - Código de criação da "ROLE" Diretores e suas permissões

Para a criação dos diretores optamos por conceder o acesso a tudo na base de dados visto que o diretor de uma agência detetive, em um cenário real, irá precisar de todas essas informações para gestão de sua agência, agindo como um administrador.

- Cliente

```
-- Criação de uma função para os clientes
CREATE ROLE 'clientes';

-- Garantir certas permissões para estes
-- Permissão para adicionar denúncias
GRANT INSERT ON Denuncia TO 'clientes';

-- Permissão para ver os seus casos
GRANT SELECT ON Caso TO 'clientes';

-- Permissão para ver os seus dados
GRANT SELECT ON Cliente TO 'clientes';
-- Permissão para atualizar os seus dados
GRANT UPDATE ON Cliente TO 'clientes';

-- Permissão para ver centros de operações disponíveis
GRANT SELECT ON Centro TO 'clientes';
```

Figura 11 - Código da criação da "ROLE" dos clientes e suas permissões

Para a criação dos clientes foi permitido a inserção de dados à tabela 'Denuncia' visto que uma denúncia é gerada por um cliente, a visualização à tabela 'Caso', 'Cliente' e 'Centro' visto que um cliente, em um cenário real, poderá eventualmente necessitar dessas informações. Entretanto nessas tabelas apenas pode atualizar a tabela dos 'Clientes' onde ele pode atualizar suas informações pessoais.

- Agente

```
-- Criação de uma função para os agentes
CREATE ROLE 'agentes';
-- Garantir certas permissões para estes
-- Permissão para ver casos
GRANT SELECT ON Caso TO 'agentes';
-- Permissão para atualizar casos
GRANT UPDATE ON Caso TO 'agentes';

-- Permissão para ver os seus dados
GRANT SELECT ON Agente TO 'agentes';
-- Permissão para atualizar os seus dados
GRANT UPDATE (nome, data_nasc, email, pass, concelho, cod_postal, rua, centro_id, tele) ON Agente TO 'agentes';
-- Permissão para ver dados dos clientes
GRANT SELECT (nome, email, concelho, cod_postal, rua) ON Cliente TO 'agentes';
GRANT SELECT ON Tele_cliente TO 'agentes';

-- Permissão para ver evidencias
GRANT SELECT ON Evidencia TO 'agentes';
-- Permissão para inserir evidencias
GRANT INSERT ON Evidencia TO 'agentes';
-- Permissão para ver suspeitos
GRANT SELECT ON Suspeito TO 'agentes';
-- Permissão para atualizar suspeitos
GRANT UPDATE ON Suspeito TO 'agentes';
-- Permissão para inserir suspeitos
GRANT INSERT ON Suspeito TO 'agentes';

-- Permissão para ver centros de operações disponíveis
GRANT SELECT ON Centro TO 'agentes';
```

Figura 12 - Código de Criação de "ROLE" dos agentes e suas permissões

Finalmente, na criação dos agentes, foi concedido o acesso às seguintes tabelas:

- 'Caso': acesso para visualizar e atualizar informações dos casos.
- 'Agente': acesso para visualizar suas informações e atualizar as suas informações.
- 'Cliente': acesso para visualizar algumas informações de contato do cliente que iniciou o processo.
- 'Evidência': acesso para visualizar e inserir novos registos.
- 'Suspeito': acesso para visualizar, inserir e atualizar informações.
- 'Centro': acesso apenas para visualizar.

5.3 Povoamento da Base de Dados

Com um conjunto Inserções (“INSERT”) e de procedimentos criados (que nos permitirão obedecer às restrições definidas anteriormente relativas à participação na caracterização dos relacionamentos) no **Anexo 7**, começamos a povoar a nossa base de dados. O código de povoamento pode ser visto no **Anexo 4**.

Como exemplo, os centros podem ser inseridos na tabela ‘Centro’ utilizando o seguinte comando:

```
INSERT INTO Centro (nome_direcao, concelho, rua, cod_postal, tele)
VALUES
('Orlando', 'Braga', 'Rua da Universidade do Minho', '4710-057', '+351253601100'),
('Vasco', 'Lisboa', 'Rua da Universidade de Lisboa', '1649-014', '+351217998000'),
('André', 'Faro', 'Rua da Universidade do Algarve', '8005-140', '+351289800100'),
('Diana', 'Aveiro', 'Rua da Universidade de Aveiro', '3810-193', '+351234370310'),
('Marisa', 'Bragança', 'Rua do Instituto Politécnico de Bragança', '5300-253', '+3512733
```

Figura 13 - Código de inserção de Centros

De forma semelhante, foram inseridos dados na tabela ‘Agente’, ‘Denuncia’, ‘Tipo_de_caso’, ‘Evidencias’.

Para a inserção de clientes, utilizamos o procedimento armazenado ‘addNovoCliente’ (que insere um cliente com um número de telemóvel já obrigatoriamente associado) presente no **Anexo 7**:

```
-- Inserção de clientes
CALL addNovoCliente('Pedro Fernandes', 'pedro@sapo.pt', '213', 'Lisboa', '1000-003', 'Rua do Milénio, 5', '+351217998000')
CALL addNovoCliente('Sara Lima', 'sara@gmail.com', '41241', 'Porto', '4000-004', 'Avenida Pasteriana, 15', '+351229980000')
CALL addNovoCliente('Joana Silva', 'joanasil@hotmail.com', 'joana', 'Coimbra', '3000-005', 'Rua das Flores, 8'
CALL addNovoCliente('Miguel Pereira', 'miguel.pereira@yahoo.com', 'miguel', 'Faro', '8000-006', 'Avenida do
CALL addNovoCliente('Ana Costa', 'anocas@gmail.com', 'anocas', 'Braga', '4700-007', 'Rua da Universidade, 20'
CALL addNovoCliente('Rui Oliveira', 'ruioliveira@outlook.com', 'ruir', 'Lisboa', '1000-008', 'Rua Nova, 30',
CALL addNovoCliente('Cátia Mendes', 'catalinamendes@gmail.com', 'catalia', 'Porto', '4000-009', 'Travessa do Sol
```

Figura 14 - Código de inserção de Clientes

Na mesma maneira, para adicionar casos, utilizamos o procedimento armazenado ‘addNovoCaso’ (o código para o procedimento vai ser explicado no 5.8) presente no **Anexo 7**, que garante a associação do caso com uma denúncia, um agente e um tipo de caso.

Além do último, também fizemos uso do procedimento armazenado ‘addNovoSuspeito’ para adicionar suspeitos e associá-los a casos.

Para associar mais um tipo de caso a um caso existente (o mesmo é permitido com as tabelas ‘Agente_responsavel_caso’ e ‘Caso_apresenta_suspeito’):

```
INSERT INTO Caso_pertence_tipo  
VALUES ('2', '1'); -- Associar mais um tipo roubo
```

Figura 15 - Código de associação de um Tipo com um Caso

Além da inserção manual de dados, desenvolvemos uma aplicação em C para inserir dados a partir de arquivos CSV na base de dados. Esta aplicação é capaz de validar os dados antes de inseri-los, garantindo que as regras de integridade sejam mantidas.

A aplicação estabelece uma conexão com a base de dados Agencia utilizando o usuário ‘program’, que foi criado especificamente para essa tarefa. Este usuário tem permissões para inserir dados em todas as tabelas do esquema Agencia.

A aplicação lê os dados dos arquivos CSV, valida-os e os insere nas tabelas especificadas. O código pode ser visto no **Anexo 8**. Um pequeno exemplo de “connection creation”:

```
MYSQL *con = mysql_init(NULL);  
  
if (con == NULL)  
{  
    fprintf(stderr, "mysql_init() failed\n");  
    exit(1);  
}  
  
if (mysql_real_connect(con, SERVER, USER, PASSWORD, DATABASE, 0, NU
```

Figura 16 - Excerto de código do programa de inserção

Atualmente, a aplicação está configurada para inserir dados apenas na tabela de clientes. No entanto, ela pode ser melhorada para inserir dados em outras tabelas, cada uma com sua própria lógica de validação. Isso permitiria uma maior flexibilidade e capacidade de gerenciamento de dados na base de dados da Agência.

5.4 Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)

Centro

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
nome_direção	VARCHAR(40)	42
Concelho	VARCHAR(20)	22
Rua	VARCHAR(50)	52
Código_postal	CHAR(8)	8
Telemóvel	CHAR(13)	13
Total	...	141

Tabela 9 - Cálculo de espaço da tabela Centro

Agente

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Nome	VARCHAR(30)	32
Cargo	VARCHAR(15)	17
Data_nasc	DATE	3
Email	VARCHAR(50)	52
Password	VARCHAR(15)	17
Concelho	VARCHAR(20)	22
Código_postal	CHAR(8)	8
Rua	VARCHAR(50)	52
Centro_id	INT	4
Telemóvel	CHAR(13)	13
Total	...	224

Tabela 10 - Cálculo de espaço da tabela Agente

Cliente

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Nome	VARCHAR(30)	32
Email	VARCHAR(50)	52
Password	VARCHAR(15)	17
Concelho	VARCHAR(20)	22
Código_postal	CHAR(8)	8
Rua	VARCHAR(50)	52
Total	...	187

Tabela 11 - Cálculo de espaço da tabela Cliente

Denúncia

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Estado_atual	VARCHAR(15)	17
Concelho	VARCHAR(20)	22
Código_postal	CHAR(8)	8
Rua	VARCHAR(50)	52
Data_denúncia	DATE	3
Descrição	VARCHAR(150)	152
Cliente_id	INT	4
Total	...	262

Tabela 12 - Cálculo de espaço da tabela Denúncia

Caso

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Data_registro	DATE	3
Data_final	DATE	3
Estado_atual	VARCHAR(15)	17
Relatório	VARCHAR(300)	302
Denúncia_id	INT	4
Total	...	333

Tabela 13 - Cálculo de espaço da tabela Caso

Tipo de caso

Atributos	Tipo de dados	Tamanho (bytes)
Numero	INT	4
Nome	VARCHAR(30)	32
Total	...	36

Tabela 14 - Cálculo de espaço da tabela Tipo de caso

Evidencia

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Descrição	VARCHAR(150)	152
Data_recolha	DATE	3
Caso_id	INT	4
Total	...	163

Tabela 15 - Cálculo de espaço da tabela Evidencia

Suspeito

Atributos	Tipo de dados	Tamanho (bytes)
Id	INT	4
Nome	VARCHAR(30)	32
Descrição	VARCHAR(150)	152
Total	...	188

Tabela 16 - Cálculo de espaço da tabela Suspeito

Telemóvel do cliente

Atributos	Tipo de dados	Tamanho (bytes)
Telemóvel_cliente	CHAR(13)	13
Cliente_id	INT	4
Total	...	17

Tabela 17 - Cálculo de espaço da tabela Telemóvel do cliente

Agente responsável por caso

Atributos	Tipo de dados	Tamanho (bytes)
Agente_id	INT	4
Caso_id	INT	4
Total	...	8

Tabela 18 - Cálculo de espaço da tabela Agente responsável por caso

Caso apresenta suspeito

Atributos	Tipo de dados	Tamanho (bytes)
Caso_id	INT	4
Suspeito_id	INT	4
Total	...	8

Tabela 19 - Cálculo de espaço da tabela Caso apresenta suspeito

Caso pertence a tipo

Atributos	Tipo de dados	Tamanho (bytes)
Caso_id	INT	4
Tipo_número	INT	4
Total	...	8

Tabela 20 - Cálculo de espaço da tabela Caso pertence a tipo

Assim concluímos que o tamanho total da nossa base de dados, com apenas 1 registo por tabela, seria 1575 bytes.

Usando como base uma estimativa de 1 ano de operação da agência tiramos os seguintes valores. 5 centros, 6 agentes, 8 clientes, 8 telemóveis de cliente, 8 denúncias, 8 tipos de casos, 6 casos, 5 suspeitos, 6 evidências, 10 instâncias de agentes responsáveis por casos, 5 instâncias de casos que apresentam suspeitos e 7 instâncias de casos pertencentes a um tipo, com estes valores o tamanho seria:

$$5*141 + 6*224 + 8*187 + 8*17 + 8*262 + 8*36 + 6*333 + 5*188 + 6*163 + 10*8 + 5*8 + 7*8 = 10157 \text{ bytes ao fim de um ano.}$$

Para estimar a taxa de crescimento anual, faremos algumas suposições baseadas em expectativas razoáveis para a expansão da agência: Com a expansão e maior visibilidade, esperamos um aumento de 30% no número de novos clientes e denúncias por ano, passando de 8 para 11. Com base nisso assumimos um crescimento anual também de 30% no número de novos casos, de 6 para 8. Cada novo caso é esperado gerar em média 1 suspeito e 1 evidência, esperando assim 8 novos suspeitos e 8 novas evidências no próximo ano. Com este aumento de casos é esperado também um aumento de 20% no número de agentes, resultando em mais um agente no próximo ano. Partindo destes números, a base de dados iria aumentar em: $11*187 + 11*262 + 8*333 + 8*188 + 8*163 + 1*224 = 10635$ bytes no ano seguinte, dando assim um total de 20792 bytes $\approx 20,8$ kB .

5.5 Definição e caracterização de vistas de utilização em SQL

```
--  
-- Criação de uma vista, onde temos todos os agentes e os seus respetivos casos não terminados  
CREATE VIEW CasosEmProgressoPorAgente AS  
SELECT A.id AS agente_id,  
        A.nome AS agente_nome,  
        C.id AS caso_id,  
        C.data_registro AS data_inicio,  
        C.relatorio AS relatorio  
FROM Agente AS A  
INNER JOIN Agente_responsavel_caso AS ARC ON A.id = ARC.agente_id  
INNER JOIN Caso AS C ON ARC.caso_id = C.id  
WHERE C.estado_atual = 'Em progresso'  
ORDER BY caso_id ASC;  
  
SELECT * FROM CasosEmProgressoPorAgente;  
-- Drop da Vista  
DROP VIEW CasosEmProgressoPorAgente;
```

Figura 17 - Código da Vista "CasosEmProgressoPorAgente"

Na **Figura 13** temos a codificação de uma vista cuja funcionalidade é a criação de uma lista dos agentes atualmente na base de dados e seus respetivos casos ainda a serem terminados, para facilitar a procura por todos os casos em progresso, juntamente com os respetivos agentes associados. Todos códigos correspondentes estarão presentes no **script** enviado no **Anexo 5**.

5.6 Tradução das interrogações do utilizador para SQL

Tendo em conta as árvores de interrogações geradas na **Secção 4** e outras interrogações associadas aos requisitos de manipulação da **Secção 2**, convertemos para código **SQL** (em procedimentos e funções) os respetivos requisitos inicialmente definidos. Este código pode ser também acessado no **Anexo 5**, onde temos todas as “queries” relacionadas com os requisitos.

A. Contar quantas denúncias os clientes submeteram – RM04

```
-- Funcão de contar denuncias feitas por certos clientes.
DELIMITER $$

CREATE FUNCTION contarDenunciasDoCliente(
    cliente INT
)
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE nDenuncias INT;

    SELECT COUNT(*) INTO nDenuncias
    FROM Denuncia
    WHERE Denuncia.cliente_id = cliente;

    RETURN nDenuncias;
END $$

DELIMITER ;

-- Usar a função -> Número de Denúncias de cada cliente
SELECT id, nome, contarDenunciasDoCliente(id) AS total_denuncias
FROM Cliente; -- WHERE Cliente.id = '1'; Caso queiramos só as denúncias do cliente com id 1
-- Remover Função
DROP FUNCTION contarDenunciasDoCliente;
```

Figura 18 - Código SQL da 1^a Interrogação

B. Listar quais agentes estão associados a um determinado caso – RM12

```
-- Enumerar informação dos agentes responsáveis por um caso - RM12
-- Procedimento - Tabela com Informação dos agentes do caso
DELIMITER $$

CREATE PROCEDURE verAgentesDoCaso(
    IN caso INT
)
BEGIN
    SELECT Agente.* FROM Agente
    INNER JOIN Agente_responsavel_caso ON Agente.id = Agente_responsavel_caso.agente_id
    WHERE Agente_responsavel_caso.caso_id = caso;
END $$

DELIMITER ;
|
-- Chamada do procedimento
CALL verAgentesDoCaso(1);
-- Drop do procedimento
DROP PROCEDURE verAgentesDoCaso;
```

Figura 19 - Código SQL da 2^a Interrogação

C. Contar número de agentes dos diversos centros – RM20

```
-- Enumerar/Contar agentes em cada centro de operações da agência - RM20
-- Função - Conta os agentes de um determinado centro
DELIMITER $$

CREATE FUNCTION contarAgentesCentro(
    centro INT
)
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE total_agentes INT;

    SELECT COUNT(id) INTO total_agentes FROM Agente
    WHERE centro_id = centro
    GROUP BY Agente.id;

    RETURN total_agentes;
END $$
DELIMITER ;

-- Uso da função para contar os agentes em cada centro
SELECT Centro.id,
    Centro.nome_direcao,
    contarAgentesCentro(Centro.id) AS numero_agentes
FROM Centro;
-- Drop Função
DROP FUNCTION contarAgentesCentro;
```

Figura 20 - Código SQL da 3^a Interrogação

D. Listar todos os casos resultantes de denúncias de um cliente – RM03

```
-- Procedimento - Visualizar casos associados a um determinado agente ou cliente - RM03
-- Do Cliente
DELIMITER $$

CREATE PROCEDURE verCasosDeCliente(
    IN cliente INT
)
BEGIN
    SELECT Caso.* FROM
        Caso INNER JOIN Denuncia ON Caso.denuncia_id = Denuncia.id
    WHERE Denuncia.cliente_id = cliente;
END $$

DELIMITER ;
|
-- Chamada do procedimento
CALL verCasosDeCliente(1);
-- Drop do procedimento
DROP PROCEDURE verCasosDeCliente;
```

Figura 21 - Código SQL da 4ª Interrogação

5.7 Indexação do Sistema de Dados

Além dos índices que iremos abordar nesta secção, na criação de uma tabela, são automaticamente gerados índices para as colunas com restrição de “PRIMARY KEY” e “UNIQUE”, que permitem um melhor desempenho em operações de procura e consulta (Por exemplo “SELECT”), apesar de não serem tão vantajosas para operações de criação, inserção ou atualização (Por exemplo “INSERT”).

Tendo em conta essas vantagens e desvantagens, além dos índices automaticamente associados, decidimos criar índices que melhorassem de um modo geral a experiência dos utilizadores da Base de Dados.

Como a tabela ‘Cliente’ irá arrecadar muitos registos de clientes e a direção da agência pretenderá manter o contacto ativo com os clientes, principalmente por e-mail, decidimos criar um índice para o mesmo, adicionalmente com a restrição de único, visto que cada cliente terá um respetivo e-mail único associado.

```
-- Criacao de um índice único sobre o atributo "email" da tabela "Cliente"
-- Unicidade Garantida e melhor desempenho em consultas (Pior nas inserções, atualizações, etc)
CREATE UNIQUE INDEX idxEMail
    ON Cliente (email);
```

Figura 22 - Criação do índice idxEMail associado ao "email" de "Cliente"

Isto trará mais rapidez no processo de procura que envolva o email do cliente, caso algum cliente tenha enviado um e-mail e quiserem achar o cliente associado na

base de dados, ou caso queiram enviar um email para todos os clientes e ter uma lista completa dos mesmos, por exemplo.

Outro índice que achamos interessante criar está relacionado à data do registo das denúncias, visto que a tabela ‘Denuncia’ terá igualmente muitos registas e as procura por datas específicas ou ordenação pelas mesmas serão utilizadas frequentemente.

```
-- Criacao de um indice sobre o atributo "data_denuncia" da tabela "Denuncia"
CREATE INDEX idxDataDenuncia
    ON Denuncia (data_denuncia);
```

Figura 23 - Criação do índice idxDataDenuncia associado à "data_denuncia" de "Denuncia"

Estes índices podem ser vistos no **Anexo 5**.

5.8 Implementação de procedimentos, funções e gatilhos

Além dos procedimentos e funções apresentadas na **Secção 5.6**, temos, no **Anexo 5 e 6**, mais alguns exemplos, juntamente com um gatilho (“Trigger”).

A. Exemplo de Procedimento com Transação “addNovoCaso”

```
-- Procedimento de adição de um caso, associado a uma denuncia, agente e tipo
CREATE PROCEDURE addNovoCaso(
    IN denuncia_id INT,
    IN agente_id INT,
    IN tipo_numero INT
)
```

Figura 24 - Criação e Input do procedimento "addNovoCaso"

A criação deste procedimento tem como objetivo facilitar a inserção de um novo caso, tendo em conta o identificador da denúncia respetiva e associando um agente e um tipo de caso, obedecendo à presença obrigatória (participação total) num caso destes últimos, já estabelecida anteriormente.

Tal como podemos seguir na próxima **Figura 25**, primeiramente é declarada a variável para controlo de erros antes de começar a transação, seguindo-se das respetivas inserções do registo do caso (com o respetivo identificador da denúncia dada no input; os restantes atributos são os “DEFAULT”, como as datas, relatório e o seu identificador de caso que é auto-incrementado), atualizando o estado da respetiva denúncia para agora “Registado” e associando o respetivo agente e tipo, com a inserção dos registos na tabela gerada através do relacionamento do caso e agente/tipo. De seguida é feita o controlo de erros durante a transação (se algo correu mal, desfaz as alterações; se correu bem, confirma as mesmas).

```

BEGIN
    DECLARE ErroTransacao BOOL DEFAULT 0;
    DECLARE novo_caso_id INT;
    DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET ErroTransacao = 1;
    START TRANSACTION;

    -- Inserir novo caso
    INSERT INTO Caso (denuncia_id) VALUES (denuncia_id);
    SET novo_caso_id = LAST_INSERT_ID();

    -- Atualizar estado denúncia
    UPDATE Denuncia SET estado_atual = "Registado" WHERE id = denuncia_id;

    -- Associar agente ao caso
    INSERT INTO Agente_responsavel_caso (agente_id, caso_id) VALUES (agente_id, novo_caso_id);

    -- Associar tipo de caso ao caso
    INSERT INTO Caso_pertence_tipo (caso_id, tipo_numero) VALUES (novo_caso_id, tipo_numero);

    IF ErroTransacao THEN
        ROLLBACK; -- Desfazer as operações realizadas.
        SELECT 'Erro ao adicionar novo caso.' AS Message;
    ELSE
        COMMIT; -- Confirmar as operações realizadas.
        SELECT 'Caso adicionado com sucesso.' AS Message;
    END IF;
END $$
```

Figura 25 - Código de execução do procedure "addNovoCaso"

B. Exemplo de Função “contarCasosResolvidosAgente”

A seguinte função da **Figura 26** foi criada para facilitar o trabalho da direção, nomeadamente no controlo dos agentes, contando o número de casos resolvidos para os mesmos, de forma a posteriormente reconhecer o trabalho dos mesmos com promoções, prémios, etc.

Esta função, recebendo um “id” de um agente irá devolver o número de casos dados como “Encerrado” na sua coluna “estado” de um determinado agente, através do “COUNT” de registos de casos encerrados, dos quais o agente faz parte, dado pelo “INNER JOIN” e posterior “WHERE”.

```
-- -----  
-- Num determinado momento, saber o nº de casos resolvidos para um dado agente - RN06  
-- Função - Conta o Número de casos resolvidos de um agente  
DELIMITER $$  
CREATE FUNCTION contarCasosResolvidosAgente(  
    agente INT  
)  
RETURNS INT  
DETERMINISTIC  
BEGIN  
    DECLARE nCasos INT;  
  
    SELECT COUNT(*)  
    INTO nCasos FROM  
    Caso INNER JOIN Agente_responsavel_caso ON Caso.id = Agente_responsavel_caso.caso_id  
    WHERE Agente_responsavel_caso.agente_id = agente AND Caso.estado_atual = 'Encerrado';  
  
    RETURN nCasos;  
END $$  
DELIMITER ;  
  
-- Uso da função para tabela com o nome do agente escolhido e os casos resolvidos por esse agente  
SELECT Agente.id, Agente.name, contarCasosResolvidosAgente(Agente.id) AS casos_resolvidos  
FROM Agente  
ORDER BY casos_resolvidos DESC; -- Do agente com mais casos resolvidos para o agente com menos casos resolvidos  
-- Drop Função  
DROP FUNCTION contarCasosResolvidosAgente;
```

Figura 26 - Código da função "contarCasosResolvidosAgente" e exemplos de uso

C. Exemplo de Gatilho “atualizarCargoAgente”

Para atualização de cargo automático dos Agentes baseado no número de casos resolvidos de cada um, foi criado um gatilho (**Figura 27**) que faz uso da anterior função da **Figura 26**. Isto facilitará a premiação imediata dos agentes após o encerramento de um caso que o englobe. O uso deste gatilho dependerá da situação da agência e obviamente não substituirá o controlo manual da direção sobre os agentes. Os números do limite são meramente exemplares, para facilitar testes sobre o gatilho.

Relativamente ao funcionamento do mesmo, ele será chamado após uma atualização do Caso, mais especificamente na atualização do seu estado para Encerrado, fazendo o uso do “NEW” e “OLD” para verificar se houve alteração do mesmo dentro do gatilho. De seguida, iremos atualizar a tabela do Agente, apenas dos respectivos agentes associados ao caso (na condição “WHERE id IN”), mudando o seu cargo ou mantendo o mesmo dependendo do seu número de casos resolvidos e respetivo cargo anterior.

```
-- Trigger - Após atualização de um caso para encerrado, irá atualizar o cargo dos agentes
-- caso os mesmos atinjam o limite de casos resolvidos estabelecidos para promoção
-- Limites alteráveis: 5 -> Veterano, 1-> Perito

DELIMITER $$

CREATE TRIGGER atualizarCargoAgente
AFTER UPDATE ON Caso
FOR EACH ROW
BEGIN
    IF NEW.estado_atual = 'Encerrado' AND OLD.estado_atual <> 'Encerrado' THEN

        UPDATE Agente
        SET cargo = CASE
            WHEN contarCasosResolvidosAgente(Agente.id) > 0 AND cargo = 'Novato' THEN 'Perito'
            WHEN contarCasosResolvidosAgente(Agente.id) > 4 AND cargo = 'Perito' THEN 'Veterano'
            ELSE cargo -- Mantém o cargo atual
        END
        WHERE id IN (
            SELECT Agente.id
            FROM Agente_responsavel_caso
            WHERE Agente_responsavel_caso.caso_id = NEW.id
        );

    END IF;
END$$
DELIMITER ;
```

Figura 27 - Código do gatilho "atualizarCargoAgente"

6. Conclusões e Trabalho Futuro

Durante este projeto, realizamos um trabalho detalhado e abrangente na construção da base de dados para a agência de detetives "Secret Story". Na primeira fase, focamos na análise das necessidades operacionais e de gestão da informação da agência, com o objetivo de desenvolver um sistema sólido e bem integrado que suportasse eficientemente o processamento de casos, a gestão de agentes, clientes, evidências, suspeitos e outros elementos críticos para as atividades da agência.

O processo de modelação começou com o levantamento detalhado dos requisitos, resultando num modelo conceitual que identificou as principais entidades, atributos e relacionamentos necessários para representar o domínio da agência de detetives. De seguida, avançamos para a elaboração do modelo lógico, que serviu de base para a implementação física da base de dados. Durante a modelagem conceptual e lógica, mantivemos sempre em mente os requisitos iniciais e o propósito da base de dados, realizando validações frequentes para garantir que os requisitos eram seguidos.

Na segunda fase do projeto, focamos na implementação física da base de dados em SQL. Usamos os modelos conceptuais e lógicos definidos na primeira fase para criar a infraestrutura de informação que a agência necessitava. O sistema foi implementado com sucesso, e todos os dados foram populados conforme o previsto, incluindo centros, agentes, clientes, telemóveis de clientes, denúncias, tipos de casos, casos, suspeitos e evidências.

Ao longo do projeto, foram identificados diversos aspectos positivos e negativos. Entre os aspectos positivos, destaca-se a aderência aos requisitos funcionais e operacionais definidos inicialmente, garantindo a eficácia no suporte às atividades da agência. A modelagem conceptual e lógica forneceu uma base teórica sólida, que facilitou a implementação física da base de dados em SQL.

Com a base de dados implementada, identificamos algumas áreas para melhorias futuras. Desenvolver ferramentas automatizadas para facilitar a inserção e atualização de dados poderia reduzir a carga de trabalho manual e minimizar erros. Implementar medidas adicionais de segurança para proteger os dados sensíveis, especialmente as informações confidenciais sobre clientes, casos e evidências. Melhorar os tipos de dados como por exemplo, aumentar o tamanho máximo das passwords dos clientes.

Em resumo, estamos confiantes de que a base de dados desenvolvida para a agência de detetives "Secret Story" está bem alinhada com os objetivos estabelecidos e pronta para sustentar eficazmente todas as atividades operacionais da agência. O sistema implementado oferece uma infraestrutura de informação robusta e funcional, que facilitará a gestão de investigações, casos e a tomada de decisões estratégicas. Com as melhorias sugeridas, esperamos aumentar ainda mais a eficiência e eficácia do sistema, garantindo que ele continue a atender plenamente às necessidades da agência de detetives.

Siglas e Acrónimos

Para facilitar a escrita do relatório, foram utilizadas as seguintes siglas:

- **ST** – Nome da agência de Detetives (*Secret Story*);
- **BD** – Base de Dados;
- **ER** – Entidade-Relacionamento;
- **SQL** - *Structured Query Language*;
- **DBMS** - *Database Management System*;

Bibliografia

As referências bibliográficas que nos ajudaram no decorrer deste projeto foram:

- Connolly, T., Begg, C., *Database Systems: A Practical Approach to Design, Implementation, and Management*, Addison-Wesley, 6^a Global Edition, 2015. (Acessado online em 2024 no site [https://www.cherrycreekeducation.com/bbk/b/Pearson Database Systems A Practical Approach to Design Implementation and Management 6th Global Edition 1292061189.pdf](https://www.cherrycreekeducation.com/bbk/b/Pearson%20Database%20Systems%20A%20Practical%20Approach%20to%20Design%20Implementation%20and%20Management%206th%20Global%20Edition%201292061189.pdf))

Anexos

- **Anexo 1** (Reunião_SecretStory.pdf) – Parte da ata da reunião com os diretores e subdiretores, sobre os requisitos para a base de dados da agência;
- **Anexo 2** (ST_Creates) – Script SQL com as criações da base de dados e tabelas;
- **Anexo 3** (ST_Permissions) – Script SQL com criações de users, roles e suas permissões;
- **Anexo 4** (ST_Sets) - Script SQL com povoamento da base de dados;
- **Anexo 5** (ST_Queries) – Script SQL com interrogações/queries dadas por procedimentos, gatilhos, funções e vistas;
- **Anexo 6** (ST_Indexes) – Script SQL com índices criados
- **Anexo 7** (ST_ActionsExamples) – Script SQL com possíveis ações de povoamento, modificação e gerência da base de dados;
- **Anexo 8** (Pasta InsertProgram) – Pasta com o programa desenvolvido em C de inserção de clientes na Base de Dados